UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/709,522	05/11/2004	Krishna Mohan ITIKARLAPALLI	ORCL-003	3521

51121 7590 08/08/2007

LAW FIRM OF NAREN THAPPETA
158, PHASE ONE PALM MEADOWS, RAMAGUNDANAHALLI
AIRPORT WHITEFIELD ROAD
BANGALORE, 560043
INDIA

EXAMINER
SANDERS, AARON J

ART UNIT	PAPER NUMBER
2168	

MAIL DATE	DELIVERY MODE
08/08/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/709,522	ITIKARLAPALLI ET AL.	
	Examiner	Art Unit	
	Aaron Sanders	2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 June 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *17 October 2006* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 5 June 2007 has been entered.

Specification

The abstract of the disclosure is objected to because it contains the title of the invention. The sheet or sheets presenting the abstract may not include other parts of the application or other material. See 37 C.F.R. 1.72.

Claim Objections

As per claim 10, "roll back" should be spelled "rollback".

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 10 and 22-24 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not

described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention. Specifically, there is no mention in the specification of how a programmer specifies different or custom rollback procedures. To the contrary, it appears that all rollbacks simply automatically undo the transactions in reverse order, see [0019].

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-24 are rejected under 35 U.S.C. 102(b) as being anticipated by Raz, U.S. 5,701,480.

As per claims 1-24, Raz teaches:

1. A method of implementing atomic transactions in a system, said method comprising:

requesting in a program logic a transaction identifier for an atomic transaction, wherein said program logic is contained in a user program designed by a programmer (See e.g. col. 7, line 57 to col. 8, line 25, "Local transactions are committed upon an explicit request from the local concurrency control mechanism" and col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 13, line 46 – col. 14, line 19, "The scheduling of operations for the

transactions is typically performed by a multi-tasking or multi-processing operating system program that services a transaction queue" which show that Raz's method is implemented in programming logic written by a programmer and used by a user);

generating said transaction identifier in a transaction manager in response to said requesting (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'");

specifying in said program logic a plurality of combinations for execution in a sequential order (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "FIG. 4A shows three different possibilities for the scheduling of a first global transaction having a write operation and a second global transaction having a conflicting read operation"), wherein each of said plurality of combinations contains said transaction identifier (See e.g. Fig. 7 where, see col. 20, lines 44-64, "The transaction list includes a linked list of transaction identification numbers 106"), a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "To obtain consistent results, the present invention permits conflicting operations of two global transactions to be scheduled in a selected order… Inconsistent scheduling possibilities, such as the possibility (b) in FIG. 4A, are prohibited by aborting a conflicting transaction when a selected global transaction is committed" where the claimed "task procedure" is the referenced "scheduling" and the claimed "rollback procedure" is the referenced "abort");

executing said task procedures in said sequential order (See e.g. Drawing Description, "FIGS. 24A and 24B together comprise a flowchart of a procedure for fetching a desired record using the pointers of the data structure of FIG. 4" and Figs. 24A-B);

keeping track of said rollback procedures in said transaction manager (See e.g. Fig. 5A where, see col. 18, lines 37-53, "To provide an interface for conducting an atomic commitment protocol for scheduling global transactions, digital computer 20 also includes a transaction manager (TM) 92"); and

executing said rollback procedures in a reverse order of said sequential order if said atomic transaction is to be aborted, wherein said rollback procedures are identified according to said keeping (See e.g. col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions").

2. The method of claim 1, wherein said transaction identifier is unique to each of the atomic transactions (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'").

3. The method of claim 1, wherein said keeping comprises storing data representing said rollback procedures in a stack (See e.g. col. 19, lines 51-59, "the transaction scheduler responds to an interrupt by removing the context of the interrupted transaction from the processor stack of the digital computer… The context includes the value of the program counter which points to the interrupted memory location in the transaction program").

4. The method of claim 3, wherein said stack is stored in a memory (See e.g. col. 2, lines 7-24, "the operating system typically provides an established set of memory management

procedures that can be invoked or called from an application program to define a 'recovery unit'", where the "stack" in the reference is part of the "recovery unit").

5. The method of claim 1, further comprising examining a status returned by execution of one of said task procedures and performing said aborting if said status indicates an error (See e.g. col. 20, lines 44-64 and col. 63, lines 45-64, "a flag R indicating whether preparation of the transaction has been completed and the transaction is ready to be committed" and "the entire before-image log file for the failed process is scanned backwards to recover and un-do the effects of a failed transaction for the failed process" respectively).

6. The method of claim 1, wherein said aborting is performed asynchronously (See e.g. col. 91, line 63 to col. 92, line 6, "Later, asynchronously, if T is committed by the AC protocol, abort all the transactions in the set $ABORT_{ECO}(T)$" where T is a transaction, see col. 85, lines 43-49).

7. A computer readable medium carrying one or more sequences of instructions representing a program logic for execution on a system, said program logic implementing an atomic transaction, wherein execution of said one or more sequences of instructions by one or more processors contained in said system causes said one or more processors to perform the actions of:

requesting an identifier for said atomic transaction (See e.g. col. 7, line 57 to col. 8, line 25, "Local transactions are committed upon an explicit request from the local concurrency control mechanism");

setting a variable to equal said identifier (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'" and Fig. 7);

specifying a plurality of combinations for execution (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "FIG. 4A shows three different possibilities for the scheduling of a first global transaction having a write operation and a second global transaction having a conflicting read operation"), wherein each of said plurality of combinations contains said transaction identifier (See e.g. Fig. 7 where, see col. 20, lines 44-64, "The transaction list includes a linked list of transaction identification numbers 106"), a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "To obtain consistent results, the present invention permits conflicting operations of two global transactions to be scheduled in a selected order… Inconsistent scheduling possibilities, such as the possibility (b) in FIG. 4A, are prohibited by aborting a conflicting transaction when a selected global transaction is committed" where the claimed "task procedure" is the referenced "scheduling" and the claimed "rollback procedure" is the referenced "abort"); and

aborting said atomic transaction by specifying said identifier associated with an abort procedure to cause said rollback procedures to be executed (See e.g. col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions"),

wherein said program logic, including said plurality of combinations, are contained in a user program designed by a programmer (See e.g. col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 13, line 46 – col. 14, line 19, "The scheduling of operations for the transactions is typically performed by a multi-tasking or multi-processing operating system program that services a transaction queue" which show that Raz's method is implemented in programming logic written by a programmer and used by a user).

8. The computer readable medium of claim 7, wherein said specifying comprises including each of said plurality of combinations in a single procedure call (See e.g. col. 2, lines 25-53, "it is desirable to distribute the operations in a transaction among multiple processors or processes in a computing system" where, in order to distribute the processes, a single function call is made in the "computing system" passing the "transaction" to be processed).

9. The computer readable medium of claim 7, further comprising examining a status returned by execution of one of said task procedures and performing said aborting if said status indicates an error (See e.g. col. 20, lines 44-64 and col. 63, lines 45-64, "a flag R indicating whether preparation of the transaction has been completed and the transaction is ready to be committed" and "the entire before-image log file for the failed process is scanned backwards to recover and un-do the effects of a failed transaction for the failed process" respectively).

10. A computer readable medium carrying one or more sequences of instructions for supporting implementation of an atomic transaction in a system, wherein execution of said one

or more sequences of instructions by one or more processors contained in said system causes said one or more processors to perform the actions of:

generating an identifier for said atomic transaction (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'");

receiving a plurality of combinations for execution (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "FIG. 4A shows three different possibilities for the scheduling of a first global transaction having a write operation and a second global transaction having a conflicting read operation"), wherein each of said plurality of combinations contains said transaction identifier (See e.g. Fig. 7 where, see col. 20, lines 44-64, "The transaction list includes a linked list of transaction identification numbers 106"), a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "To obtain consistent results, the present invention permits conflicting operations of two global transactions to be scheduled in a selected order… Inconsistent scheduling possibilities, such as the possibility (b) in FIG. 4A, are prohibited by aborting a conflicting transaction when a selected global transaction is committed" where the claimed "task procedure" is the referenced "scheduling" and the claimed "rollback procedure" is the referenced "abort");

executing said task procedures (See e.g. Drawing Description, "FIGS. 24A-B together comprise a flowchart of a procedure for fetching a desired record using the pointers of the data structure of FIG. 4" and Figs. 24A-B); and

executing said rollback procedures in response to receiving an abort request (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "Inconsistent scheduling possibilities, such as the possibility (b) in FIG. 4A, are prohibited by aborting a conflicting transaction when a selected global transaction is committed"),

wherein each of said plurality of combinations is received from a corresponding user program (See e.g. Fig. 5A where, see col. 18, lines 14-36, "Global and local transactions are initiated, for example, by application programs 90") and wherein the roll back procedure received from a first user program is different from the roll back procedure received from a second user program (See e.g. col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions" where the transactions that will be aborted are different for different user program initiated transactions).

11. The computer readable medium of claim 10, wherein said task procedures are executed in an execution order and corresponding rollback procedures are executed in a reverse order of said execution order (See e.g. col. 2, lines 7-24, "The recovery unit consists of program statements between a 'START' statement and a 'COMMIT' statement. All of the statements in the 'recovery unit' must be completed before the memory records modified by the statements in the recovery unit are made available for subsequent processing… The statements in the 'recovery unit' specify operations in a single 'transaction'").

12. The computer readable medium of claim 11, further comprising storing data indicating that said rollback procedures are to be executed in said reverse order to abort said atomic transaction (See e.g. col. 2, lines 7-24, "The recovery unit consists of program statements

between a 'START' statement and a 'COMMIT' statement. All of the statements in the 'recovery unit' must be completed before the memory records modified by the statements in the recovery unit are made available for subsequent processing… The statements in the 'recovery unit' specify operations in a single 'transaction'").

13. The computer readable medium of claim 12, wherein said identifier is generated to be unique for each atomic transaction (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'").

14. The computer readable medium of claim 12, wherein said data is represented in the form of a stack (See e.g. col. 19, lines 51-59, "the transaction scheduler responds to an interrupt by removing the context of the interrupted transaction from the processor stack of the digital computer… The context includes the value of the program counter which points to the interrupted memory location in the transaction program").

15. The computer readable medium of claim 14, wherein said stack is stored in a memory (See e.g. col. 2, lines 7-24, "the operating system typically provides an established set of memory management procedures that can be invoked or called from an application program to define a 'recovery unit'", where the "stack" in the reference is part of the "recovery unit").

16. A computer system comprising:

a memory storing a plurality of instructions (See e.g. col. 2, lines 7-24, "the operating system typically provides an established set of memory management procedures that can be invoked or called from an application program to define a 'recovery unit.' The recovery unit consists of program statements between a 'START' statement and a 'COMMIT' statement");

and

a processing unit coupled to said memory and executing said plurality of instructions to support implementation of atomic transactions in a programming environment, said processing unit being operable to (See e.g. Fig. 1, Central Processing Unit 21 and Volatile Random Access Memory 22):

request in a program logic a transaction identifier for an atomic transaction (See e.g. col. 7, line 57 to col. 8, line 25, "Local transactions are committed upon an explicit request from the local concurrency control mechanism");

generate said transaction identifier in a transaction manager in response to said requesting (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'");

specify in said program logic a plurality of combinations for execution in a sequential order (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "FIG. 4A shows three different possibilities for the scheduling of a first global transaction having a write operation and a second global transaction having a conflicting read operation"), wherein each of said plurality of combinations contains said transaction identifier (See e.g. Fig. 7 where, see col. 20, lines 44-64, "The transaction list includes a linked list of transaction identification numbers 106"), a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure (See e.g. Figs. 4A-B where, see col. 17, lines 13-37, "To obtain consistent results, the present invention permits conflicting operations of two global transactions to be scheduled in a selected order… Inconsistent scheduling possibilities, such as the possibility (b) in FIG. 4A, are prohibited by aborting a conflicting transaction when a selected global transaction is committed"

where the claimed "task procedure" is the referenced "scheduling" and the claimed "rollback procedure" is the referenced "abort");

execute said task procedures in said sequential order (See e.g. Drawing Description, "FIGS. 24A-B together comprise a flowchart of a procedure for fetching a desired record using the pointers of the data structure of FIG. 4" and Figs. 24A-B);

keep track of said rollback procedures in said transaction manager (See e.g. Fig. 5A where, see col. 18, lines 37-53, "To provide an interface for conducting an atomic commitment protocol for scheduling global transactions, digital computer 20 also includes a transaction manager (TM) 92"); and

execute said rollback procedures in a reverse order of said sequential order if said atomic transaction is to be aborted, wherein said rollback procedures are identified according to said keeping (See e.g. col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions"),

wherein said program logic is contained in a user program designed by a programmer (See e.g. col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 13, line 46 – col. 14, line 19, "The scheduling of operations for the transactions is typically performed by a multi-tasking or multi-processing operating system program that services a transaction queue" which show that Raz's method is implemented in programming logic written by a programmer and used by a user).

17. The computer system of claim 16, wherein said transaction identifier is unique to each of the atomic transactions (See e.g. col. 2, lines 25-53, "To identify the transaction being performed, the transaction is typically assigned a unique 'transaction identification number'").

18. The computer system of claim 16, wherein said processing unit is operable to store data representing said rollback procedures in a stack to perform said keep (See e.g. col. 19, lines 51-59, "the transaction scheduler responds to an interrupt by removing the context of the interrupted transaction from the processor stack of the digital computer… The context includes the value of the program counter which points to the interrupted memory location in the transaction program").

19. The computer system of claim 18, wherein said stack is stored in a memory (See e.g. col. 2, lines 7-24, "the operating system typically provides an established set of memory management procedures that can be invoked or called from an application program to define a 'recovery unit'", where the "stack" in the reference is part of the "recovery unit").

20. The computer system of claim 16, wherein said processing unit is further operable to examine a status returned by execution of one of said task procedures and to perform said aborting if said status indicates an error (See e.g. col. 20, lines 44-64 and col. 63, lines 45-64, "a flag R indicating whether preparation of the transaction has been completed and the transaction is ready to be committed" and "the entire before-image log file for the failed process is scanned backwards to recover and un-do the effects of a failed transaction for the failed process" respectively).

21. The computer system of claim 16, wherein said processing unit is operable to execute said rollback procedures asynchronously (See e.g. col. 91, line 63 to col. 92, line 6, "Later,

asynchronously, if T is committed by the AC protocol, abort all the transactions in the set $ABORT_{ECO}(T)$" where T is a transaction, see col. 85, lines 43-49).

22. The method of claim 1, wherein different user programs contain different rollback procedures such that each programmer can specify custom rollback procedures in the respective user program for execution in said system (See e.g. col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions" where the transactions that will be aborted are different for different user program initiated transactions).

23. The method of claim 7, wherein different user programs contain different rollback procedures such that each programmer can specify custom rollback procedures in the respective user program for execution in said system (See e.g. col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions" where the transactions that will be aborted are different for different user program initiated transactions).

24. The computer system of claim 16, wherein different user programs contain different rollback procedures such that each programmer can specify custom rollback procedures in the

respective user program for execution in said system (See e.g. col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions" where the transactions that will be aborted are different for different user program initiated transactions).

Response to Arguments

In response to Applicant's argument that the references fail to show certain features of Applicant's invention, it is noted that the features upon which Applicant relies (i.e., specifying different rollback procedures for each combination of instructions) are not recited in rejected claim 1. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). Claim 1 simply states, "specifying… a plurality of combinations for execution in a sequential order, wherein each of said plurality of combinations contains… a rollback procedure… and said rollback procedure is designed to rollback said task procedure". The claim does not require that the rollback procedures be different for each combination.

As per Applicant's argument that Raz does not teach "wherein different user programs contain different rollback procedures… for execution in said system" in claim 22, the Examiner respectfully disagrees. The Examiner cited col. 11, lines 50-59, "Turning now to FIG. 1, there is

shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions", which teaches the claimed subject matter.

In response to Applicant's argument that the references fail to show certain features of Applicant's invention, it is noted that the features upon which Applicant relies (i.e., "wherein each of said plurality of combinations is received from a corresponding user program and wherein the roll back procedure received from a first user program is different from the roll back procedure received from a second user program") are not recited in rejected claim 7. Claim 7 has not been amended to recite the above limitation.

As per Applicant's argument that Raz does not teach "receiving a plurality of combinations for execution… wherein each of said plurality of combinations is received from a corresponding user program and wherein the roll back procedure received from a first user program is different from the roll back procedure received from a second user program" in claim 10, the Examiner respectfully disagrees. The Examiner cited Fig. 5A where, see col. 18, lines 14-36, "Global and local transactions are initiated, for example, by application programs 90", which teaches receiving transactions from a corresponding user program and col. 7, lines 19-37, "recoverability is enforced by a process of cascading aborts; the aborting of a transaction requires the additional aborting of all other transactions that have read data written by aborted transactions" which shows that the transactions that will be aborted, i.e. the claimed "rollback procedure", are different for different user program initiated transactions. More specifically,

read-write transactions, i.e. the claimed "combinations", can overlap, see Figs. 4A-B. Subsequently when there is a failure or an abort request, they are aborted, i.e. rolled back, in a "cascading" or hierarchical order, which will be different for different transactions.

As per Applicant's argument that Raz does not teach "wherein said program logic is contained in a user program designed by a programmer" in claim 16, the Examiner respectfully disagrees. The Examiner cited col. 11, lines 50-59, "Turning now to FIG. 1, there is shown a block diagram generally designated 20 of a digital computer configured for transaction processing. The computer 20 includes a central processing unit 21 for executing programmed instructions" and col. 13, line 46 – col. 14, line 19, "The scheduling of operations for the transactions is typically performed by a multi-tasking or multi-processing operating system program that services a transaction queue" which show that Raz's method is implemented in programming logic written by a programmer and used by a user.

Conclusion

The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure: Gostanian et al., U.S. 5,781,910; McKeehan et al., U.S. 5,920,863; and Weedon, U.S. 6,799,188.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aaron J. Sanders whose telephone number is 571-270-1016. The examiner can normally be reached on M-Th 8:00a-5:00p.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Vo Tim can be reached on 571-272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/AJS/
Aaron J. Sanders
Examiner
30 July 2007

SRP
7/31

TIM VO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100